Compiler Options

Option	Type	Default	Description
`--allowJs`	`boolean`	`false`	Allow JavaScript files to be compiled.
`--allowSyntheticDefaultImports`	`boolean`	`module === "system"`	Allow default imports from modules with no default export. This does not affect code emit, just typechecking.
`--allowUnreachableCode`	`boolean`	`false`	Do not report errors on unreachable code.
`--allowUnusedLabels`	`boolean`	`false`	Do not report errors on unused labels.
`--alwaysStrict`	`boolean`	`false`	Parse in strict mode and emit `"use strict"` for each source file
`--baseUrl`	`string`		Base directory to resolve non-relative module names. See Module Resolution documentation for more details.
`--charset`	`string`	`"utf8"`	The character set of the input files.
`--declaration` `-d`	`boolean`	`false`	Generates corresponding `.d.ts` file.
`--declarationDir`	`string`		Output directory for generated declaration files.
`--diagnostics`	`boolean`	`false`	Show diagnostic information.
`--disableSizeLimit`	`boolean`	`false`	Disable size limitation on JavaScript project.
`--emitBOM`	`boolean`	`false`	Emit a UTF-8 Byte Order Mark (BOM) in the beginning of output files.
`--emitDecoratorMetadata`[1]	`boolean`	`false`	Emit design-type metadata for decorated declarations in source. See issue #2577 for details.
`--experimentalDecorators`[1]	`boolean`	`false`	Enables experimental support for ES decorators.
`--forceConsistentCasingInFileNames`	`boolean`	`false`	Disallow inconsistently-cased references to the same file.
`--help` `-h`			Print help message.
`--importHelpers`	`boolean`	`false`	Import emit helpers (e.g. `__extends`, `__rest`, etc..) from `tslib`
`--inlineSourceMap`	`boolean`	`false`	Emit a single file with source maps instead of having a separate file.

Option	Type	Default	Description
`--inlineSources`	`boolean`	`false`	Emit the source alongside the sourcemaps within a single file; requires `--inlineSourceMap` or `--sourceMap` to be set.
`--init`			Initializes a TypeScript project and creates a `tsconfig.json` file.
`--isolatedModules`	`boolean`	`false`	Unconditionally emit imports for unresolved files.
`--jsx`	`string`	`"Preserve"`	Support JSX in `.tsx` files: `"React"` or `"Preserve"`. See JSX.
`--jsxFactory`	`string`	`"React.createElement"`	Specify the JSX factory function to use when targeting react JSX emit, e.g. `React.createElement` or `h`.
`--lib`	`string[]`		List of library files to be included in the compilation. Possible values are: ► `ES5` ► `ES6` ► `ES2015` ► `ES7` ► `ES2016` ► `ES2017` ► `DOM` ► `DOM.Iterable` ► `WebWorker` ► `ScriptHost` ► `ES2015.Core` ► `ES2015.Collection` ► `ES2015.Generator` ► `ES2015.Iterable` ► `ES2015.Promise` ► `ES2015.Proxy` ► `ES2015.Reflect` ► `ES2015.Symbol` ► `ES2015.Symbol.WellKnown` ► `ES2016.Array.Include` ► `ES2017.object` ► `ES2017.SharedMemory` Note: If `--lib` is not specified a default library is injected. The default library injected is: ► For `--target ES5`: `DOM,ES5,ScriptHost` ► For `--target ES6`: `DOM,ES6,DOM.Iterable,ScriptHost`
`--listEmittedFiles`	`boolean`	`false`	Print names of generated files part of the compilation.

Option	Type	Default	Description
`--listFiles`	boolean	false	Print names of files part of the compilation.
`--locale`	string	*(platform specific)*	The locale to use to show error messages, e.g. en-us.
`--mapRoot`	string		Specifies the location where debugger should locate map files instead of generated locations. Use this flag if the .map files will be located at run-time in a different location than the .js files. The location specified will be embedded in the sourceMap to direct the debugger where the map files will be located.
`--maxNodeModuleJsDepth`	number	0	The maximum dependency depth to search under node_modules and load JavaScript files. Only applicable with `--allowJs`.
`--module` `-m`	string	`target === "ES6" ? "ES6" : "CommonJS"`	Specify module code generation: `"None"`, `"CommonJS"`, `"AMD"`, `"System"`, `"UMD"`, `"ES6"`, or `"ES2015"`. ► Only `"AMD"` and `"System"` can be used in conjunction with `--outFile`. ► `"ES6"` and `"ES2015"` values may not be used when targeting `"ES5"` or lower.
`--moduleResolution`	string	`module === "AMD" | "System" | "ES6" ? "Classic" : "Node"`	Determine how modules get resolved. Either `"Node"` for Node.js/io.js style resolution, or `"Classic"`. See Module Resolution documentation for more details.
`--newLine`	string	*(platform specific)*	Use the specified end of line sequence to be used when emitting files: `"crlf"` (windows) or `"lf"` (unix)."
`--noEmit`	boolean	false	Do not emit outputs.
`--noEmitHelpers`	boolean	false	Do not generate custom helper functions like `__extends` in compiled output.
`--noEmitOnError`	boolean	false	Do not emit outputs if any errors were reported.
`--noFallthroughCasesInSwitch`	boolean	false	Report errors for fallthrough cases in switch statement.
`--noImplicitAny`	boolean	false	Raise error on expressions and declarations with an implied `any` type.
`--noImplicitReturns`	boolean	false	Report error when not all code paths in function return a value.

Option	Type	Default	Description
`--noImplicitThis`	`boolean`	`false`	Raise error on `this` expressions with an implied `any` type.
`--noImplicitUseStrict`	`boolean`	`false`	Do not emit `"use strict"` directives in module output.
`--noLib`	`boolean`	`false`	Do not include the default library file (`lib.d.ts`).
`--noResolve`	`boolean`	`false`	Do not add triple-slash references or module import targets to the list of compiled files.
`--noUnusedLocals`	`boolean`	`false`	Report errors on unused locals.
`--noUnusedParameters`	`boolean`	`false`	Report errors on unused parameters.
~~`--out`~~	`string`		DEPRECATED. Use `--outFile` instead.
`--outDir`	`string`		Redirect output structure to the directory.
`--outFile`	`string`		Concatenate and emit output to single file. The order of concatenation is determined by the list of files passed to the compiler on the command line along with triple-slash references and imports. See output file order documentation for more details.
`paths`[2]	`Object`		List of path mapping entries for module names to locations relative to the `baseUrl`. See Module Resolution documentation for more details.
`--preserveConstEnums`	`boolean`	`false`	Do not erase const enum declarations in generated code. See const enums documentation for more details.
`--pretty`[1]	`boolean`	`false`	Stylize errors and messages using color and context.
`--project` `-p`	`string`		Compile a project given a valid configuration file. The argument can be an file path to a valid JSON configuration file, or a directory path to a directory containing a `tsconfig.json` file. See tsconfig.json documentation for more details.
`--reactNamespace`	`string`	`"React"`	Specifies the object invoked for `createElement` and `__spread` when targeting `"react"` JSX emit.
`--removeComments`	`boolean`	`false`	Remove all comments except copy-right header comments beginning with `/*!`

Option	Type	Default	Description
`--rootDir`	`string`	*(common root directory is computed from the list of input files)*	Specifies the root directory of input files. Only use to control the output directory structure with `--outDir`.
`rootDirs`[2]	`string[]`		List of *root* folders whose combined content represent the structure of the project at runtime. See Module Resolution documentation for more details.
`--skipDefaultLibCheck`	`boolean`	`false`	Skip type checking of default library declaration files.
`--skipLibCheck`	`boolean`	`false`	Skip type checking of all declaration files (`*.d.ts`).
`--sourceMap`	`boolean`	`false`	Generates corresponding `.map` file.
`--sourceRoot`	`string`		Specifies the location where debugger should locate TypeScript files instead of source locations. Use this flag if the sources will be located at run-time in a different location than that at design-time. The location specified will be embedded in the sourceMap to direct the debugger where the source files will be located.
`--strictNullChecks`	`boolean`	`false`	In strict null checking mode, the `null` and `undefined` values are not in the domain of every type and are only assignable to themselves and `any` (the one exception being that `undefined` is also assignable to `void`).
`--stripInternal`[1]	`boolean`	`false`	Do not emit declarations for code that has an `/** @internal */` JSDoc annotation.
`--suppressExcessPropertyErrors`	`boolean`	`false`	Suppress excess property checks for object literals.
`--suppressImplicitAnyIndexErrors`	`boolean`	`false`	Suppress `--noImplicitAny` errors for indexing objects lacking index signatures. See issue #1232 for more details.
`--target` `-t`	`string`	`"ES3"`	Specify ECMAScript target version: `"ES3"` (default), `"ES5"`, `"ES6"`/`"ES2015"`, `"ES2016"`, `"ES2017"` or `"ESNext"`. Note: `"ESNext"` targets latest supported ES proposed features.
`--traceResolution`	`boolean`	`false`	Report module resolution log messages.

Option	Type	Default	Description
`--types`	`string[]`		List of names of type definitions to include. See @types, –typeRoots and – types for more details.
`--typeRoots`	`string[]`		List of folders to include type definitions from. See @types, –typeRoots and – types for more details.
`--version` `-v`			Print the compiler's version.
`--watch` `-w`			Run the compiler in watch mode. Watch input files and trigger recompilation on changes.

- [1] These options are experimental.
- [2] These options are only allowed in `tsconfig.json`, and not through command-line switches.

Related

- Setting compiler options in `tsconfig.json` files.
- Setting compiler options in MSBuild projects.